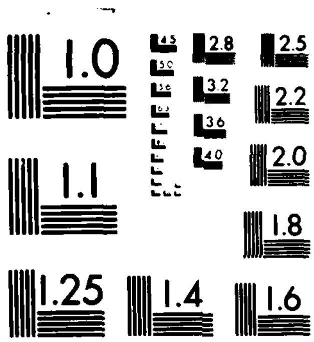
AD-A129 314 DATA VERIFICATION PROGRAMS FOR THE RSRE E-BEAM DATA FORMAT(U) ROYAL SIGNALS AND RADAR ESTABLISHMENT MALVERN (ENGLAND) B C MERRIFIELD JAN 83 RSRE-MEMO-3550 1/1

UNCLASSIFIED DRIC-BR-87379 F/G 9/2 NL

END
DATE
FILMED
7-83
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNLIMITED

RSRE
MEMORANDUM No. 3550

ROYAL SIGNALS & RADAR ESTABLISHMENT

DATA VERIFICATION PROGRAMS FOR THE RSRE E- BEAM DATA FORMAT

Author: B C Merrifield

PROCUREMENT EXECUTIVE,
MINISTRY OF DEFENCE,
RSRE MALVERN,
WORCS.

UNLIMITED

ROYAL SIGNALS AND RADAR ESTABLISHMENT

Memorandum 3550

Title: DATA VERIFICATION PROGRAMS FOR THE RSRE EBEAM DATA FORMAT

Author: B C Merrifield

Date: January 1983

SUMMARY

A number of Fortran programs designed to verify data produced for the RSRE Electron Beam Lithography Machine (EBLM-1) are described. Data files intended for exposure by EBLM-1 can be input to any of the programs depending on the level of verification required. Output is hard copy on a Servogor 281 Digital plotter or visual on a DEC VT125 graphics terminal. The programs provide an invaluable means of checking complex data structures before committing valuable machine time and materials.

RSRE MEMORANDUM NO 3550

DATA VERIFICATION PROGRAMS FOR THE RSRE EBEAM DATA FORMAT

B C Merrifield

CONTENTS

1 INTRODUCTION

The data structure for the RSRE Electron Beam Lithography Machine (EBLM-1) in P2 division has been described in detail elsewhere (Ref 1). The present memorandum describes a number of programs designed to validate that data. The programs all have similar formats and consist of data input sections and pattern 'exposure' sections whereby data patterns or lists can be 'exposed' either on a digital plotter or on the screen of a graphics display terminal.

The programs are written in Fortran 77 and have been developed on a PDP-11/44 computer under the RSX-11M operating system. The plotter is a Servogor-281 8-colour A3 digital plotter (also marketed by Calcomp and others under their own names), with automatic paper feed. The graphics terminal is a DEC VT125 terminal with REGIS.

The programs allow the designer to produce hard copy (or VDU) plots of the data at various levels in the hierarchical data structure. The lowest level is the pattern level which is the rectangle data within a single 100 μm subfield. The second level is the list level which is all the data within a 1 mm field. The third level is the chip level which is a plot of the whole data structure. An additional program PLOTWEDGE is included to provide plots of the standard data pattern for the frequency wedge.

All the plotting programs make use of a package (Ref 2) of routines purchased with the plotter. These routines are written in Macro-11 and are an

almost straightforward mapping of the plotters own commands. A further package (Ref 3) at a higher level is also available but is not used in this instance. This package (Hardware Compatible Basic Software-HCBS) is written in Fortran 4 and provides high level plotting facilities. The programs outputting to the VT125 graphics terminal use the DEC 'REGIS' graphics language.

This memorandum describes a number of programs designed to provide a hard copy record or visual verification of pattern data prepared in the EBLM-1 data format. A single example is used to illustrate the operation and output of each program. Output from the plotter is restricted to A4 size for convenience.

2 PLOTTER PROGRAMS

The plotter is assumed to be configured as a separate peripheral and not in series with the terminal, it should be switched to REMOTE and is assumed to be supplied with pens in stalls 1 and 2 and with single sheet or continuous paper. The operating system is assumed to be RSX-11M (V3.2) and the library file of plotting routines PLT281LIB is assumed to have been created as described in Ref 2. Some additional files containing routines common to all the programs are also required, they are:-

```
        PLOTRECT.FTN
        ERRRPT.FTN
        NTOCH.MAC
  and   CHPSZE.FTN
```

together with their corresponding object files.

2.1 Building the Programs

All files are assumed to be in the current user file directory. The source program filenames are as follows:

```
PLOTPATTN.FTN
PLOTLIST.FTN
PLOTCHIP.FTN
PLOTWEDGE.FTN
```

which have been compiled by:

```
F77 PLOTPATTN=PLOTPATTN
F77 PLOTLIST=PLOTLIST
F77 PLOTCHIP=PLOTCHIP
F77 PLOTWEDGE=PLOTWEDGE
```

thereby creating object files of the same names. For each file there is a corresponding command file which can be used to taskbuild the programs. These command files have the same names as the programs they build but have the extension .CMD, for example:

```
PLOTPATTN.CMD
```

which contains the following commands

```
[1,54]PLOTPATTN/FP/CP=PLOTPATTN,NTOCH,PLOTRECT,ERRRPT,
LB:[1,1]PLT281LIB/LB
/
ASG=TT3:1
//
```

This command file assigns the plotter to TT3 (the default logical unit number of the plotter is 1) and creates a checkpointable file PLOTPATTN.TSK in UIC [1,54]. The command file is invoked as follows:

```
TKB @ PLOTPATTN
```

and similarly for PLOTLIST and PLOTCHIP.

2.2 Running PLOTPATTN and PLOTLIST

The following instructions are explicitly for the program PLOTPATTN, to run PLOTLIST substitute LIST wherever PATTERN or PATTN occurs. The task files are installed as follows:

```
INS $PLOTPATTN
```

and the program run with

```
RUN PLOTPA
```

note that RUN will only accept 6 character names whereas INSTALL (INS) accepts 9 character names. The program should now respond with the message:

TWO PENS ARE REQUIRED : 1=PATTERN;2=TEXT,ETC,OK TO CONTINUE? (Y/N)

> Reply Y to continue, this delay gives the user the opportunity to check that the plotter is set up correctly and that the required coloured pens are in the holders. The pattern will be drawn with pen 1 and the text etc with Pen 2.

ENTER FILE NAME

> Enter a data file name in RSX-11M format eg DEMO.DAT followed by RETURN, to which, unless there are any errors the program will reply.
>
> DATA IN, NO OF ENTRIES = nnn
>
> where nnn is the number of data entries. If there is an error in the data the messages:
>
> ERROR,MORE THAN nnn ENTRIES
> or
> DATA READ ERROR, INPUT TERMINATED
>
> may be displayed, depending on whether the data is too large or some non-numeric character has been detected.

BRIEF PLOT?

Enter Y to eliminate some time consuming captions.

AUTO CHART ADVANCE?(Y/N)

Answer Y if plotter has automatic paper feed and continuous plotting is required. This enables multiple plots to be made without operator intervention. Answer N if single sheet plotting is required.

PLOT ALIGNMENT MARKERS? (Y/N)

Answer Y if the alignment markers are required. Note this facility is not available with PLOTCHIP.

PLOT ALL PATTERNS (THERE ARE nnn)?

Answer Y if all the patterns are required where nnn is the number of patterns. Answer N if selected patterns are required.

HOW MANY PATTERNS?

This question is only asked if the reply to 'plot all patterns?' was N.Enter the number of patterns required. Note a reply of 0 (zero) will terminate the program.

NOW ENTER PATTERN NUMBERS

If a non-zero reply was given to the previous question, now enter the numbers of the patterns required, separated by commas eg 0,5,6,7. Numbering of the patterns is in the order they are in the data file, *starting at zero.*

DO YOU WANT TO ANNOTATE?

This question is not asked if automatic chart advance has been selected. Answer Y if annotation is required on single sheet plots.

NOW ENTER TEXT HERE

This invitation will not be offered if the reply to the previous question was N.Enter lines of text terminating each line with a RETURN, input is ended by a RETURN without any text. The length of a line of text is restricted to about 50 characters and the number of lines to about 6. (These restrictions are imposed by the layout and can be extended slightly if writing outside of the plotting area is acceptable).

CHANGE PAPER, OK?

If single sheet plotting has been selected this instruction will be displayed after each plot. Enter Y to continue after changing paper. Enter N to terminate program.

PLOTPA -- STOP PLOT COMPLETE

This message indicates the successful completion of the plots. Note due to internal buffers the plotter may continue for some time after this message has been displayed.

2.3 Running PLOTCHIP

Program PLOTCHIP differs slightly from PLOTPATTN and PLOTLIST, in that additional information is output and some options are no longer applicable. The task file is installed and run exactly as before, ie

```
INS $PLOTCHIP
RUN PLOTCH
```

As before the program will respond with the messages:

TWO PENS ARE REQUIRED : 1=TEXT,ETC;2=CHIP PATTERN, OK TO CONTINUE? (Y/N)

ENTER FILE NAME

to which the responses are exactly as indicated in section 2.2. If the data is input correctly some additional information is printed ie

CHIP SIZE = nnnnnnnn.n nnnnnnnn.n

where nnnnnnnn.n denote the X and Y dimensions of the chip in microns. An additional question is now asked:

SHOW INDIVIDUAL PATTERNS?(Y/N)

Because PLOTCHIP plots the whole of the data structure including stage movements the total area to be plotted can be quite large with a consequent reduction in scale. If the size of the chip is such that individual patterns would be indistinguishable and perhaps more importantly would take an unacceptable length of time, a reply of N will result in the borders of the subfield containing a pattern being plotted rather than the pattern itself. This enables the overall structure to be shown without wasting time on details. A reply of N will also result in the message 'INDIVIDUAL PATTERNS NOT PLOTTED' being inserted on the plot.

DO YOU WANT TO ANNOTATE?

Answer Y if annotation is required.

NOW ENTER TEXT HERE

Enter lines of text as described in section 2.2.

PLOTCH -- STOP PLOT COMPLETE

Output on successful completion of the plot.

An additional feature of the Chip level plot is that the start and end positions of the plot are marked, thereby indicating the direction of stage movement.

2.4 Running PLOTWEDGE

Program PLOTWEDGE is designed to output a specific data structure and no other. The data is that built into the RSRE Electron Beam Lithography Suite (Ref.4) to produce a sequence of adjoining rectangles 5 μm wide by 20 μm high exposed at frequencies from 800 to 1600 KHz. This data is designed to produce an exposure 'wedge' the thickness of which can be measured on, for example, an Alphastep machine, to provide information on the correct exposure for a particular resist.

The only input to the program is the reply to the prompt:

TWO PENS ARE REQUIRED : 1=TEXT,GRID ETC;2=WEDGE, OK TO CONTINUE? (Y/N)

on successful completion the program will display:

PLOTWE -- STOP PLOT COMPLETE

An example of the plotter output from this program is given in figure 1.

2.5 Illustrative Examples

A single data file has been used to illustrate all the programs. The name of the file is DEMO.DAT and it is in fact the pattern used as a marker for alignment of multi-layer devices. A print-out of DEMO.DAT is given in Appendix A. The following is a copy of the corresponding computer dialogue which produced the plots shown in figures 2 to 8. Unfortunately it has not been possible to reproduce them in colour, changes in which will, therefore, only show up as lighter or darker lines.

Figure 1 demonstrates the PLOTWEDGE program. The varying height lines on the top of the rectangles are reference marks denoting 100 μm steps along the wedge. The following is a copy of the terminal log of command sequences which produced the plot. The character > is the RSX-11M MCR prompt.

```
>INS $PLOTWEDGE
>RUN PLOTWE
TWO PENS ARE REQUIRED:1=TEXT,GRID ETC;2=WEDGE OK TO CONTINUE? (Y/N)
>Y
PLOTWE -- STOP PLOT COMPLETE
```

Figures 2, 3 and 4 illustrate the PLOTPATTN program. Figure 2 shows a plot of pattern number 0 with the 'transparent' reference markers superimposed. It will be noted that the actual pattern which is in the bottom left hand corner, extends beyond the 100 μm subfield. This will not normally be the case, and is a facility not advertised for the general user.

```
>INS $PLOTPATTN
>RUN PLOTPA
TWO PENS ARE REQUIRED :1=PATTERN;2=TEXT,ETC, OK TO CONTINUE? (Y/N)
>Y
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES = 196
BRIEF PLOT? N
AUTO CHART ADVANCE?(Y/N) N
PLOT ALIGNMENT MARKERS? (Y/N) Y
PLOT ALL PATTERNS (THERE ARE 3)?N
HOW MANY PATTERNS? 1
NOW ENTER PATTERN NUMBERS 0
DO YOU WANT TO ANNOTATE? Y
NOW ENTER TEXT

FIG.2 PATTERN LEVEL WITH MARKERS

PLOTPA -- STOP PLOT COMPLETE
```

The above log shows that several blank lines have been included in the text in order to avoid the pattern which extends down into the area normally reserved for text. These lines are achieved by typing a few blanks followed by a RETURN since a RETURN on its own terminates the text input. Figures 3 and 4 were produced from the following log:

```
>RUN PLOTPA
TWO PENS ARE REQUIRED :1=PATTERN;2=TEXT,ETC,OK TO CONTINUE? (Y/N)
>Y
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES=196
BRIEF PLOT? N
AUTO CHART ADVANCE?(Y/N) N
PLOT ALIGNMENT MARKERS? (Y/N) N
PLOT ALL PATTERNS(THERE ARE 3)?N
HOW MANY PATTERNS?2
NOW ENTER PATTERN NUMBERS 0,1
DO YOU WANT TO ANNOTATE? Y
NOW ENTER TEXT

FIG.3 PATTERN LEVEL WITHOUT MARKERS,PATTERN 0.

CHANGE PAPER,OK? Y
DO YOU WANT TO ANNOTATE? (Y/N) Y
NOW ENTER TEXT

FIG.4 PATTERN LEVEL WITHOUT MARKERS,PATTERN 1.

PLOTPA -- STOP PLOT COMPLETE
```

This sequence illustrates the selection of two patterns from three, this time without markers. At the end of the first plot the program will prompt for a change of paper (unless auto chart advance is selected) before commencing the second pattern. The user is also given the opportunity to annotate the plot.

Figures 5 and 6 demonstrate the PLOTLIST program. The plots produced are now at list level and show a 1 mm field. Figure 5 is a plot of list 0 with the transparent markers included. Figure 6 is the same plot without markers and with 'brief plot' selected. The command sequence is as follows:

```
>INS $PLOTLIST
>RUN PLOTLI
TWO PENS ARE REQUIRED:1=LIST;2=TEXT,ETC,OK TO CONTINUE? (Y/N)
>Y
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES=196
BRIEF PLOT? N
AUTO CHART ADVANCE?(Y/N) N
PLOT ALIGNMENT MARKERS? (Y/N) Y
PLOT ALL LISTS(THERE ARE 1)?Y
DO YOU WANT TO ANNOTATE? Y
NOW ENTER TEXT FIG.5 LIST LEVEL WITH MARKERS.
```

```
PLOTLI -- STOP PLOT COMPLETE
>RUN PLOTLI
TWO PENS ARE REQUIRED:1=LIST;2=TEXT,ETC,OK TO CONTINUE? (Y/N)
>Y
ENTER FILE NAME DEMO.DAT
DATA IN, NO OF ENTRIES=196
BRIEF PLOT? Y
AUTO CHART ADVANCE? (Y/N) N
PLOT ALIGNMENT MARKERS? (Y/N) N
PLOT ALL LISTS(THERE ARE 1)?Y
DO YOU WANT TO ANNOTATE? Y
NOW ENTER TEXT FIG.6 LIST LEVEL BRIEF PLOT WITHOUT MARKERS.

PLOTLI -- STOP PLOT COMPLETE
```

Figures 7 and 8 illustrate the PLOTCHIP program. This time the plot is of the whole structure defined in the data. In the case of DEMO.DAT this is only an area of two square millimeters which allows a large enough scale for individual patterns to be seen as shown in Figure 7. The usual chip however will be measured in cms and Figure 8 shows the effect of electing not to show individual patterns. Here the outline of the subfield containing a pattern is drawn, this has the effect of reducing the plotting time while still giving some idea of the overall shape and enabling repetition distances to be checked. The command sequence is as follows:

```
>INS $PLOTCHIP
>RUN PLOTCH
TWO PENS ARE REQUIRED :1=TEXT,ETC;2=CHIP PATTERN, OK TO CONTINUE? (Y/N) Y
ENTER FILE NAME DEMO.DAT
DATA IN, NO OF ENTRIES = 196
CHIP SIZE = 2000.0  2000.0
SHOW INDIVIDUAL PATTERNS? (Y/N) Y
DO YOU WANT TO ANNOTATE? Y
NOW ENTER TEXT FIG.7 CHIP LEVEL (COMPLETE DATA STRUCTURE).

PLOTCH -- STOP PLOT COMPLETE

>RUN PLOTCH
TWO PENS ARE REQUIRED :1=TEXT,ETC;2=CHIP PATTERN, OK TO CONTINUE? (Y/N) Y
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES = 196
CHIP SIZE =  2000.0  2000.0
SHOW INDIVIDUAL PATTERNS? (Y/N) N
DO YOU WANT TO ANNOTATE? Y
NOW ENTER TEXT FIG.8 CHIP LEVEL, PATTERN OUTLINES ONLY

PLOTCH -- STOP PLOT COMPLETE
```

3 VT125 GRAPHICS DISPLAY PROGRAMS

The digital plotter programs described in the previous section have counterparts for the VT125 graphics terminal called VT125LIST, VT125PATT and VT125CHIP. These programs allow a quick check to be made on a data file, no hard copy is produced and so no accurate measurements or comparisons can be made. As each of these programs follow a similar format they will not be considered individually in the following description. Any differences in input or responses will be pointed out at the time.

3.1 Building the Programs

As before all files are assumed to be in the current user file directory and the source filenames are:

```
VT125LIST.FTN
VT125PATT.FTN
VT125CHIP.FTN
```

Additional files common to some or all of the programs are:

```
    VTPLOTRT.FTN
and CHPSZE.FTN
```

All these programs and subroutines are compiled to produce object files of the same names. As before, command files also with the same names contain the necessary task build instructions. These command files contain the following:

a) VT125LIST.CMD - [1,54]VT125LIST/FP/CP=VTPLOTRT,VT125LIST
b) VT125PATT.CMD - [1,54]VT125PATT/FP/CP=VTPLOTRT,VT125PATT
c) VT125CHIP.CMD - [1,54]VT125CHIP/FP/CP=VTPLOTRT,CHPSZE,VT125CHIP

and are invoked as follows:

```
>TKB @VT125LIST
```

and similarly for VT125PATT and VT125CHIP.

3.2 Running the Programs

All of the programs are installed and run in the same way, for example:

```
>INS $VT125LIST
>RUN VT125L
```

and all of them will respond with:

ENTER FILE NAME

the reply to which should be exactly as prescribed in section 2.2. Program VT125CHIP will output the following additional information:

CHIPSIZE = nnnnnn.n nnnnnn.n

where nnnnnn.n represents the actual chip dimensions in microns. All the programs output the following question:

SHADING ON? (Y/N)

answer 'Y' to fill in enclosed figures, answer 'N' for border lines only. Program VT125CHIP now asks the following question:

SHOW INDIVIDUAL PATTERNS? (Y/N)

to which the comments of section 2.3 apply. VT125CHIP will now clear the screen and begin to plot the chip. Programs VT125LIST and VT125PATT ask the further question:

PLOT ALL LISTS{PATTERNS} (THERE ARE nn)

followed by

HOW MANY LISTS{PATTERNS}?

and

NOW ENTER LIST{PATTERN}NOS

to which the responses are as described in section 2.2.

VT125LIST and VT125PATT will now begin to plot the lists or patterns requested. If several lists {patterns} are requested the plot is displayed for a few seconds before the screen is cleared and the next plot begins. While the screen is in use, scrolling is restricted to the bottom 4 lines.

3.3 Illustrative Examples

The data file DEMO.DAT, used to illustrate the plotter programs, has been used again to demonstrate the VT125 programs. The following is a copy of the computer dialogue which produced the displays shown in figs 9 to 14 which illustrate the various facilities available.

```
>INS $VT125LIST
>RUN VT125L
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES=196
SHADING ON? (Y/N) Y
PLOT ALL LISTS (THERE ARE 1)? Y
```

The above commands produced the display shown in figure 9.

```
>INS $VT125PATT
>RUN VT125P
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES=196
SHADING ON? (Y/N) Y
PLOT ALL PATTERNS (THERE ARE 3)? N
HOW MANY PATTERNS? 1
NOW ENTER PATTERN NOS 0
>RUN VT125P
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES=196
SHADING ON? (Y/N) Y
PLOT ALL PATTERNS (THERE ARE 3)? N
HOW MANY PATTERNS? 1
NOW ENTER PATTERN NOS 1
>RUN VT125P
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES=196
SHADING ON? (Y/N) Y
PLOT ALL PATTERNS (THERE ARE 3)? N
HOW MANY PATTERNS? 1
NOW ENTER PATTERN NOS 2
```

The above commands produced the displays shown in figures 10, 11 and 12. The plots could equally well have been produced all at the same time with the following commands:

```
>RUN VT125P
ENTER FILE NAME DEMO.DAT
DATA IN, NO OF ENTRIES=196
SHADING ON? (Y/N) Y
PLOT ALL PATTERNS (THERE ARE 3)? Y
```

This method was not used in order to allow plenty of time to photograph each pattern. Figures 13 and 14 were produced with the following command sequence:

```
>INS $VT125CHIP
>RUN VT125C
ENTER FILE NAME DEMO.DAT
DATA IN, NO OF ENTRIES=196
SHADING ON? (Y/N)Y
SHOW INDIVIDUAL PATTERNS? (Y/N) Y
>RUN VT125C
ENTER FILE NAME DEMO.DAT
DATA IN,NO OF ENTRIES=196
SHADING ON? (Y/N) Y
SHOW INDIVIDUAL PATTERNS? (Y/N) N
```

REFERENCES

1 V J Mifsud, B C Merrifield, A Simple data input format for an Electron-beam Lithography machine. RSRE MEMO No 3361 June 1981.

2 DEC Gesellschaft m.b.H, RSX-11M/IAS operating systems Fortran extensions for plotter type 281.

3 DEC Gesellschaft m.b.H, RSX-11M/IAS operating systems Fortran IV/Fortran IV Plus Hardware Compatible Basic Software for plotter type 281.

4 B C Merrifield, EBEAM: An operating system for particle beam lithography, RSRE MEMO NO 3512 October 1982.

APPENDIX A

Listing of demonstration data file

```
0 0 100 200 1 1 1 4 1 3 3
1 0 1
1 0 1
1 0 1
4 0 1 1 1 2 1 3 1
0 1 0 0 0  160 0
0 1 0 0 160 0 0
0 1 0 0 0 -160 0
0 1 0 0 0 0 0
3 0 1 2
0 0 0 12
2 1 4 1 6 1 8 1 2 3 8 3 2 5 8 5 2 7 4 7 6 7 8 7
0 0 1 12
3 1 5 1 7 1 2 2 8 2 2 4 8 4 2 6 8 6 3 7 5 7 7 7
0 0 2 4
3 2 7 2 3 6 7 6
5
848 1552 1648 1584
848 1392 1648 1424
848 1232 1648 1264
848 1072 1648 1104
848 912 1648 944
5
912 848 944 1648
1072 848 1104 1648
1232 848 1264 1648
1392 848 1424 1648
1552 848 1584 1648
6
1200 928 1296 1568
928 1200 1568 1296
732 732 764 1690
732 732 1690 764
1658 732 1690 1690
732 1658 1690 1690
```

FREQUENCY WEDGE ON 30-NOV-82

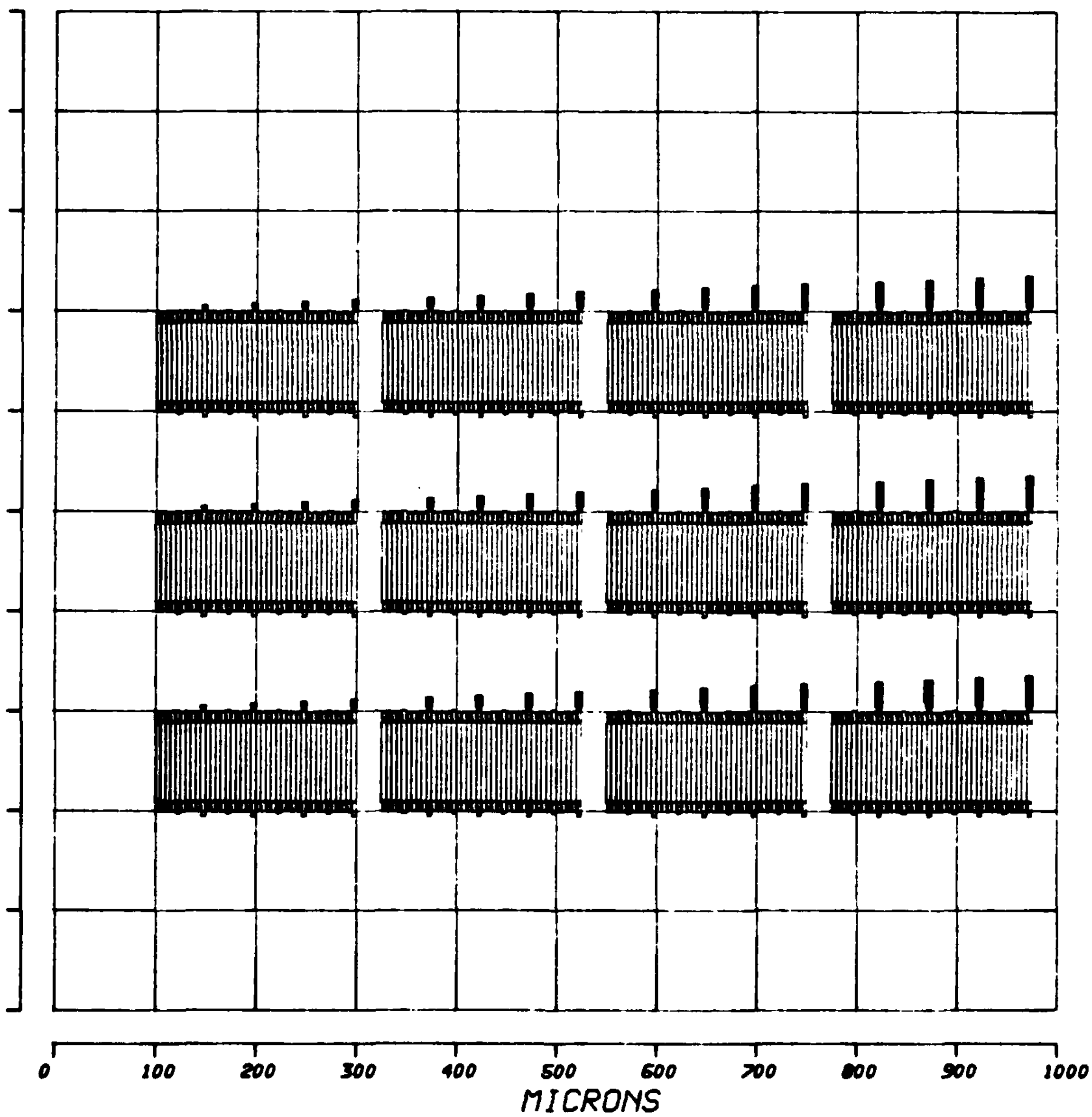


FIG. 1 FREQUENCY WEDGE

DATA FILE NAME: DEMO.DAT

ON 30-NOV-82

PATTERN NO.0

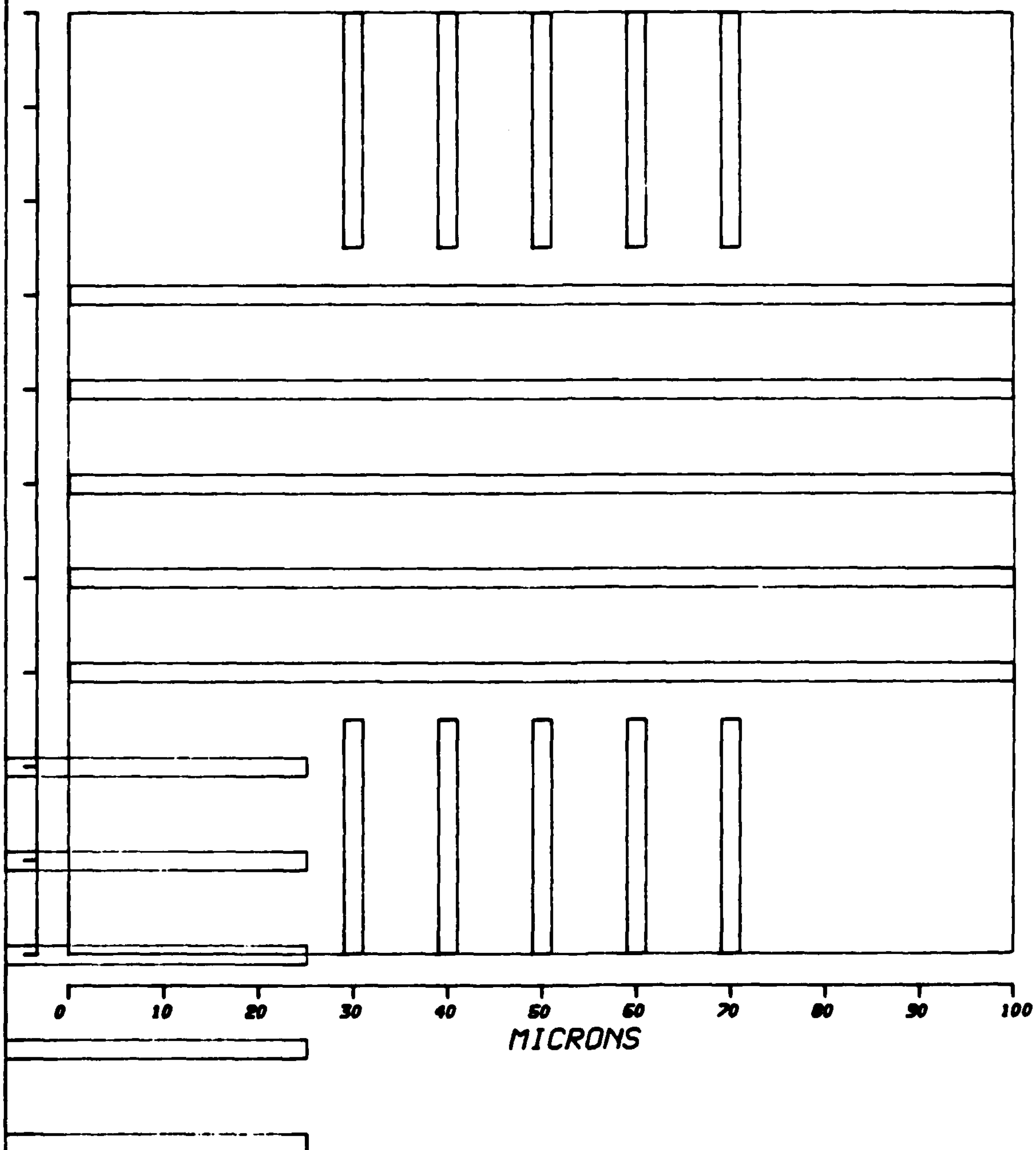


FIG.2 PATTERN LEVEL WITH MARKERS..

DATA FILE NAME: DEMO.DAT

ON 30-NOV-82

PATTERN NO.0

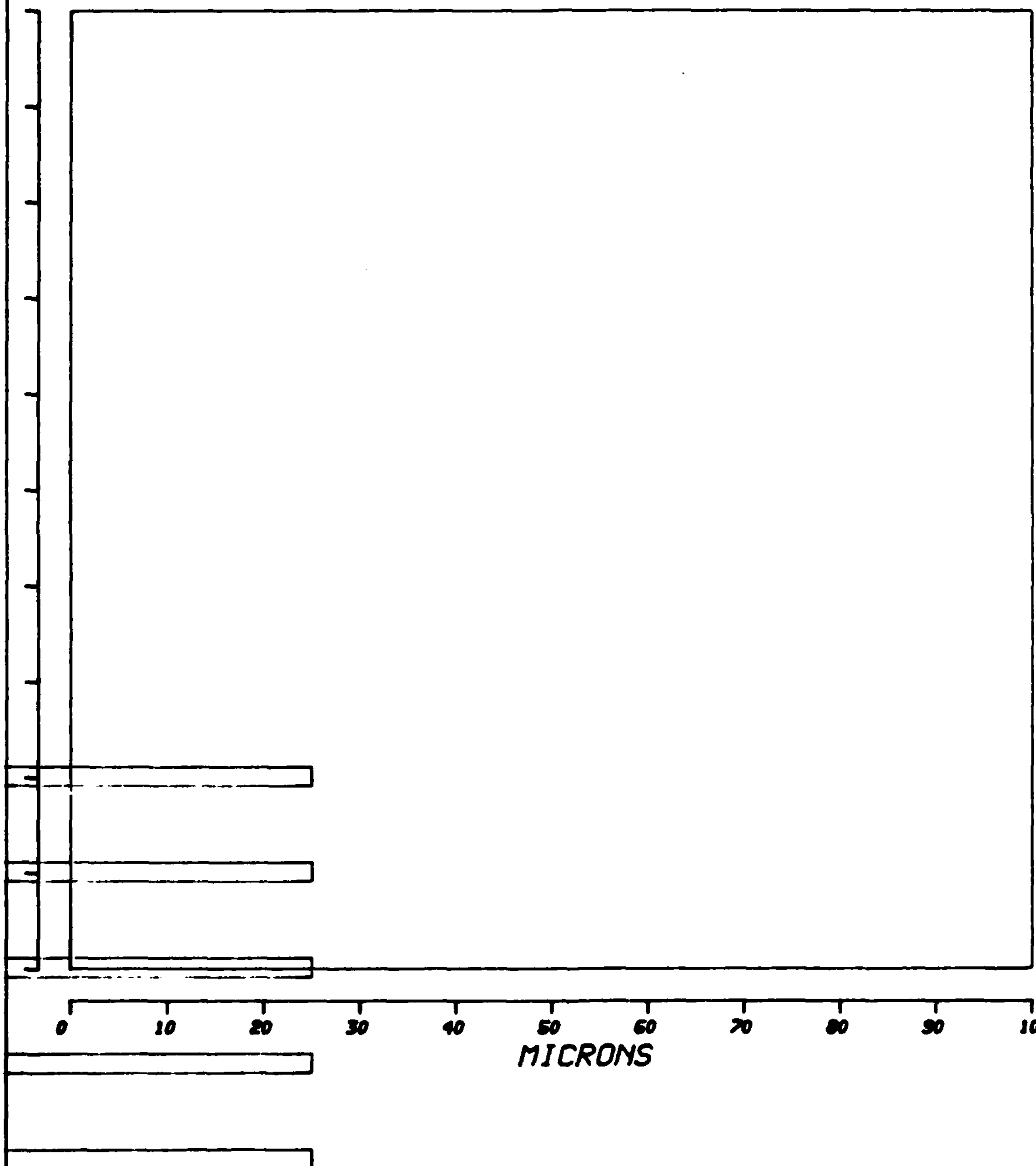


FIG.3 PATTERN LEVEL WITHOUT MARKERS, PATTERN 0.

DATA FILE NAME: DEMO.DAT

ON 30-NOV-82

PATTERN NO. 1

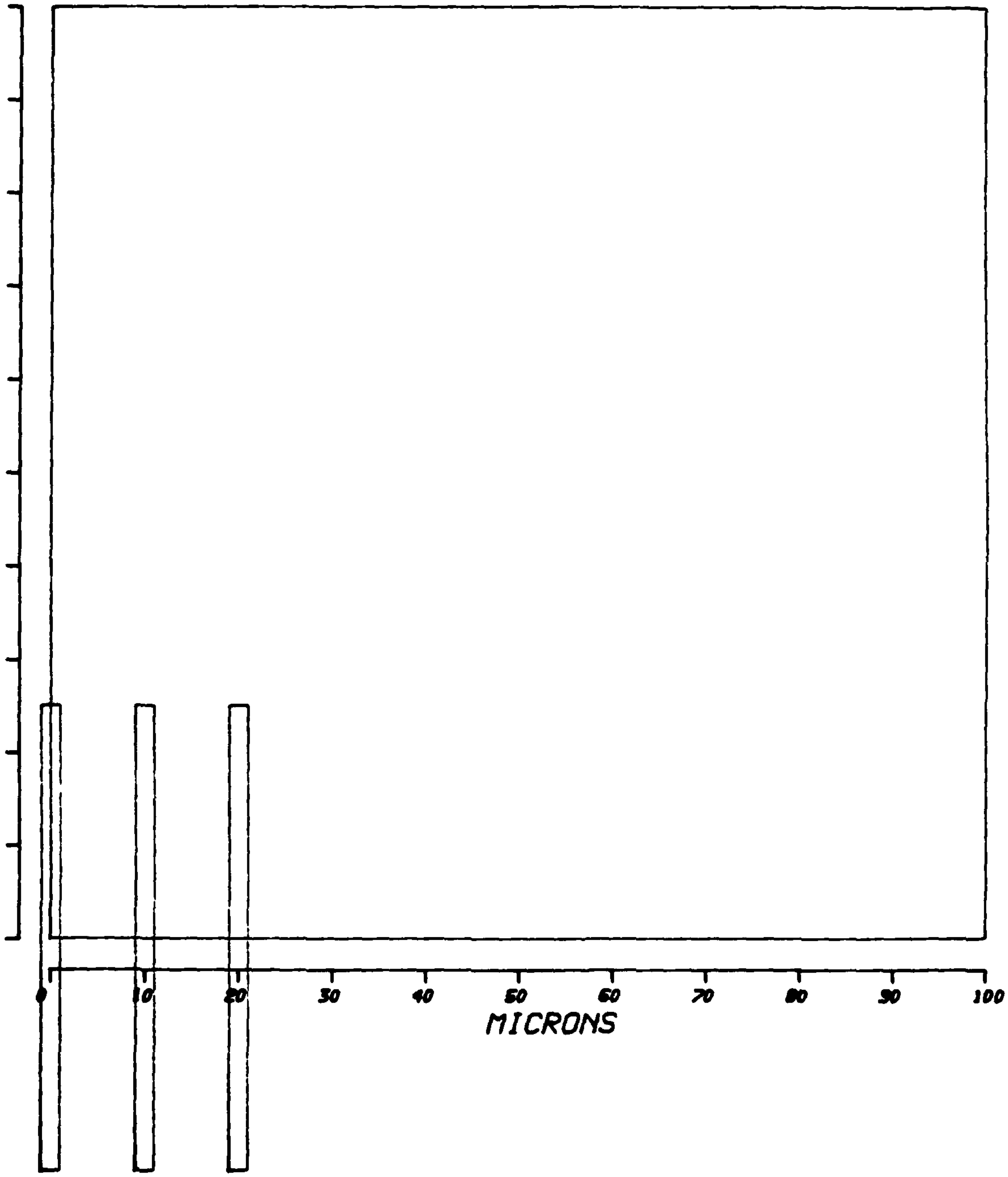


FIG. 4 PATTERN LEVEL WITHOUT MARKERS, PATTERN 1.

DATA FILE NAME: DEMO.DAT

ON 30-NOV-82

LIST NO.0

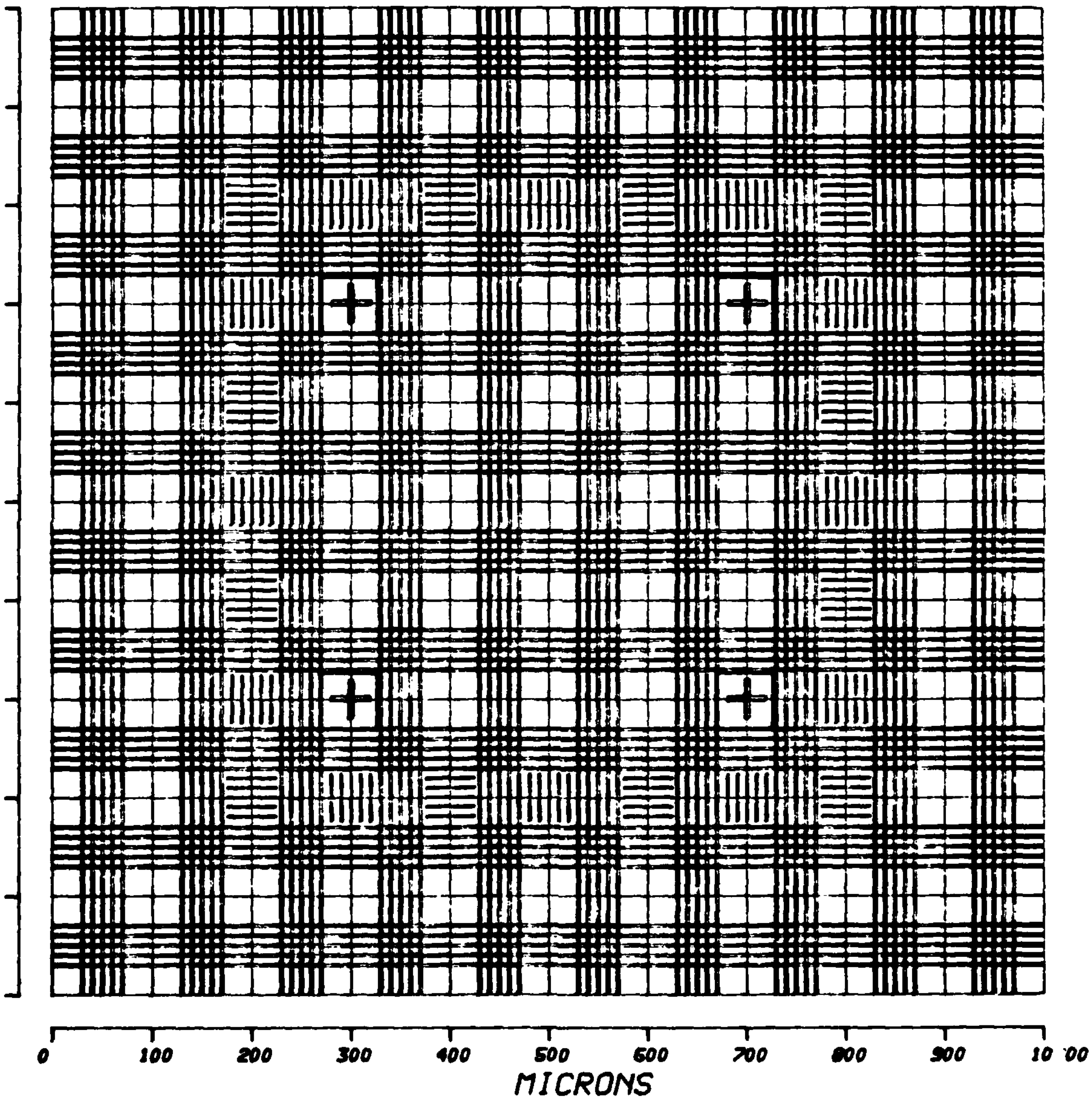


FIG.5 LIST LEVEL WITH MARKERS.

LIST NO.0

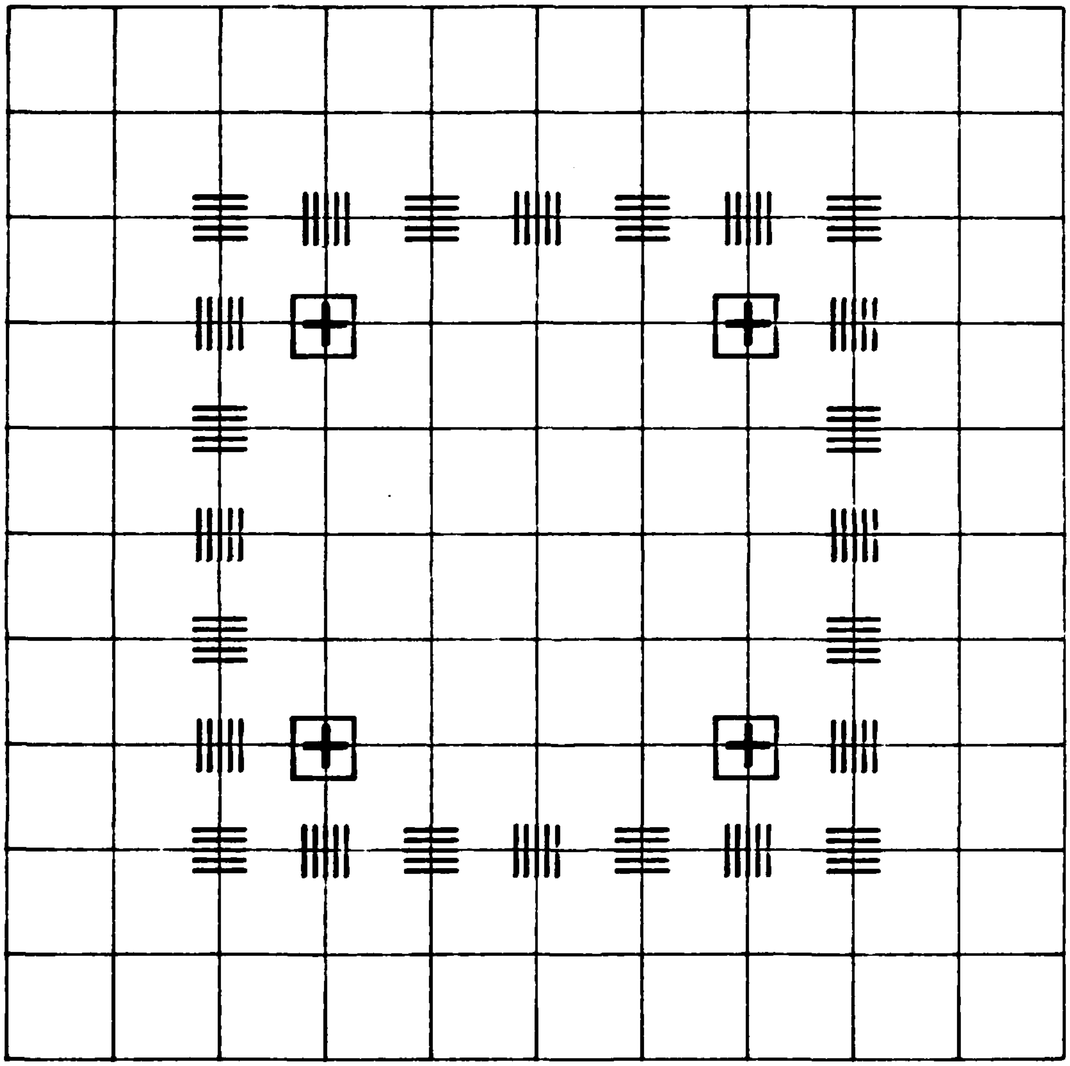


FIG.6 LIST LEVEL BRIEF PLOT WITHOUT MARKERS.

DATA FILE NAME: DEMO.DAT

ON 30-NOV-82

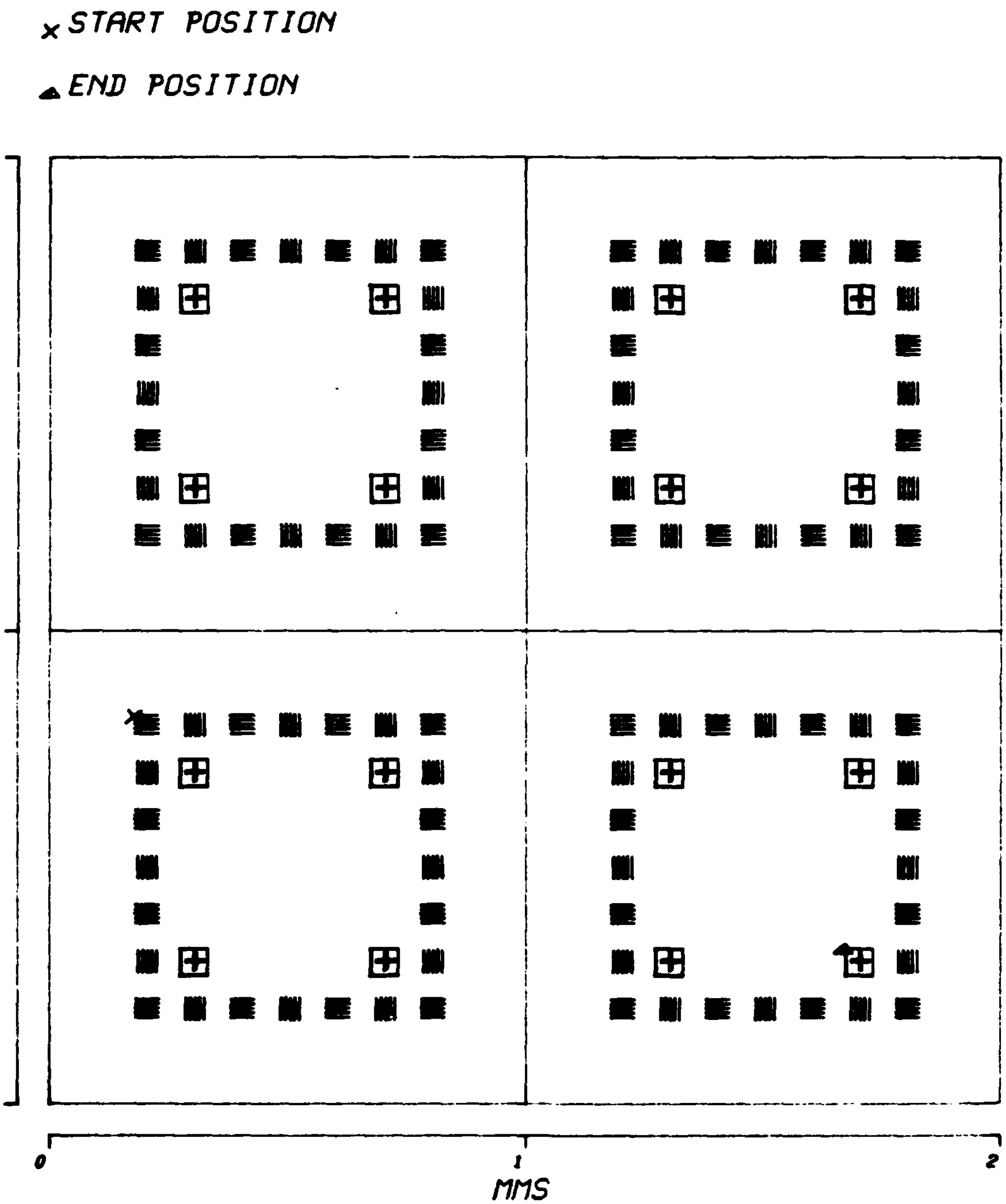


FIG.7 CHIP LEVEL (COMPLETE DATA STRUCTURE).

DATA FILE NAME: DEMO.DAT

ON 30-NOV-82

INDIVIDUAL PATTERNS NOT PLOTTED

× START POSITION

▲ END POSITION

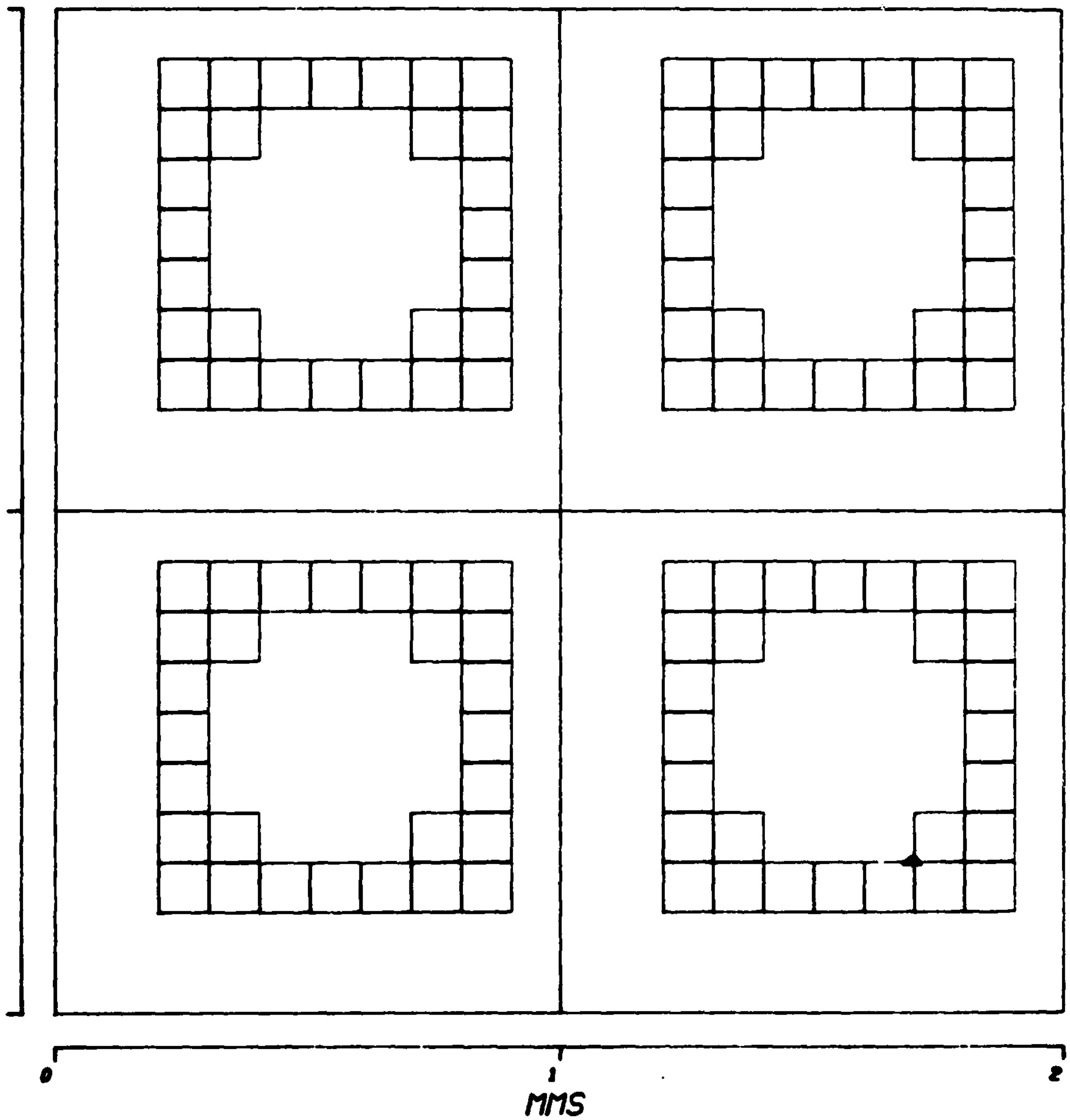


FIG.8 CHIP LEVEL, PATTERN OUTLINES ONLY

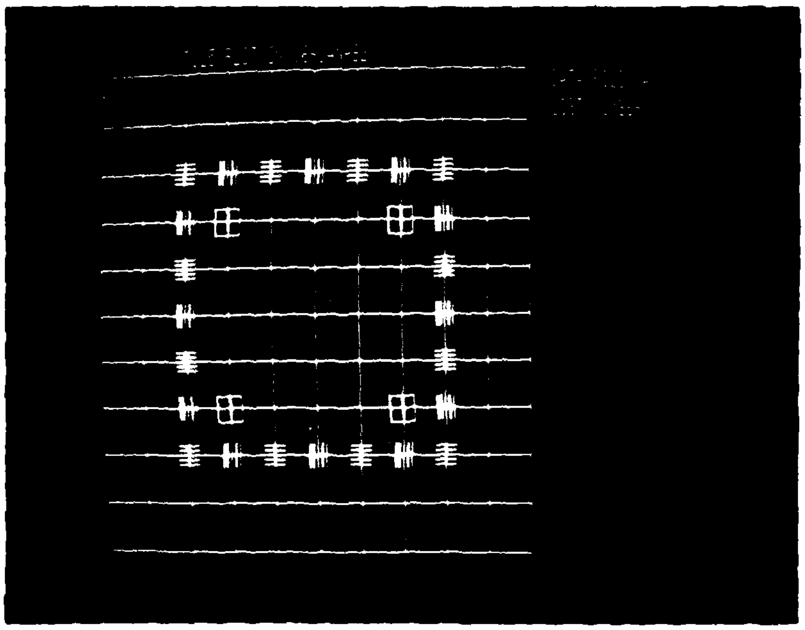


FIG 9. VT125 plot of DEMO.DAT (LIST 0)

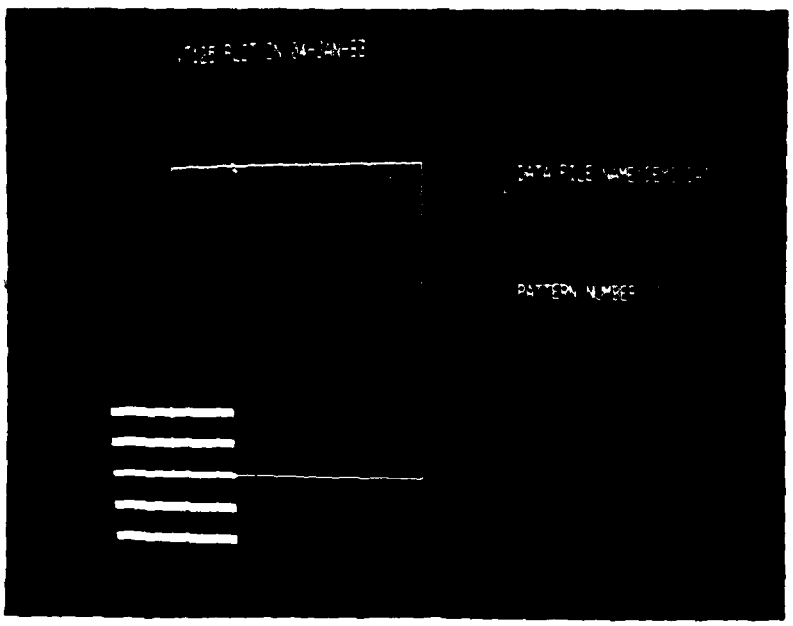


FIG 10. VT125 plot of DEMO.DAT (PATTERN 0)

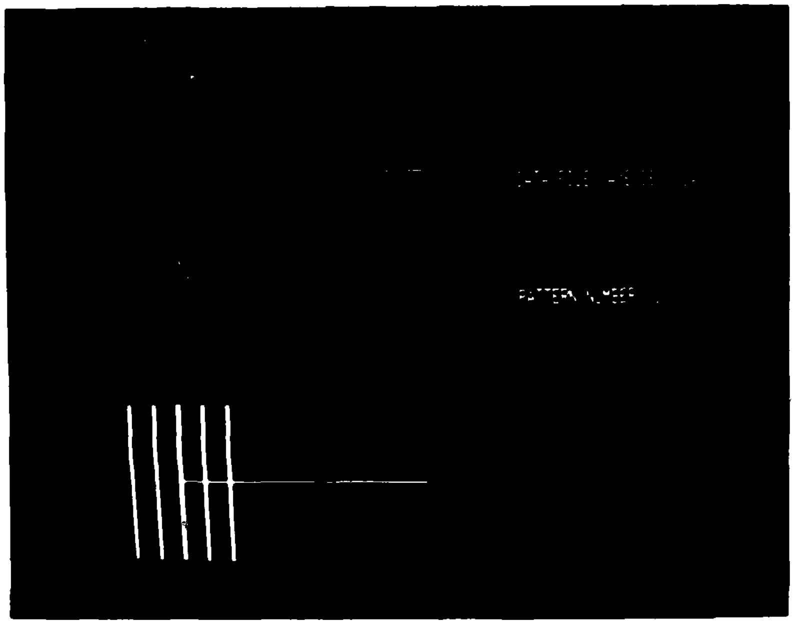


FIG 11. VT125 plot of DEMO.DAT (PATTERN 1)

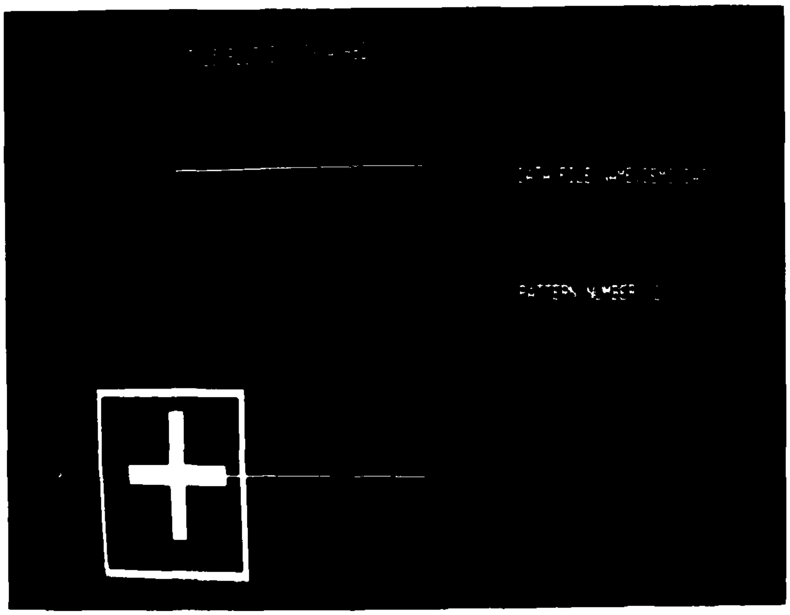


FIG 12. VT125 plot of DEMO.DAT (PATTERN 2)

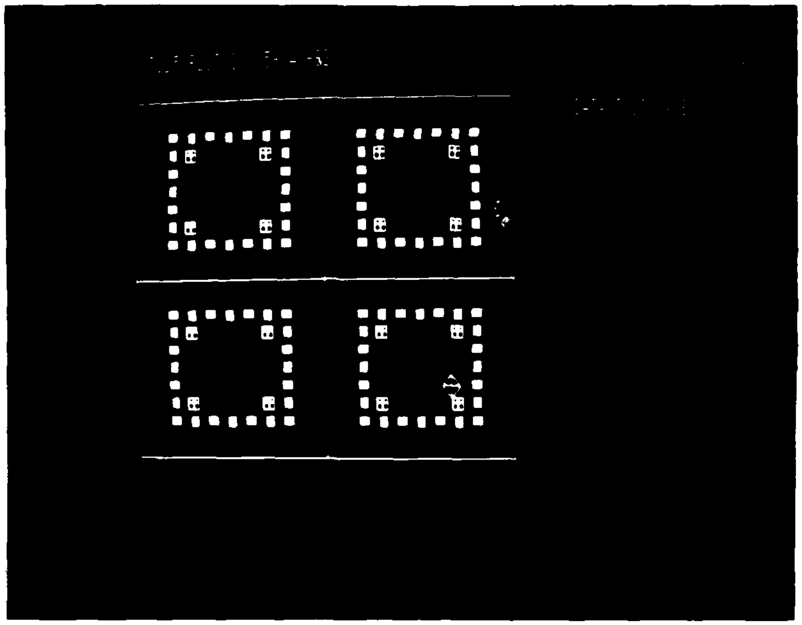


FIG 13. VT125 Chip-level plot of DEMO.DAT with individual patterns

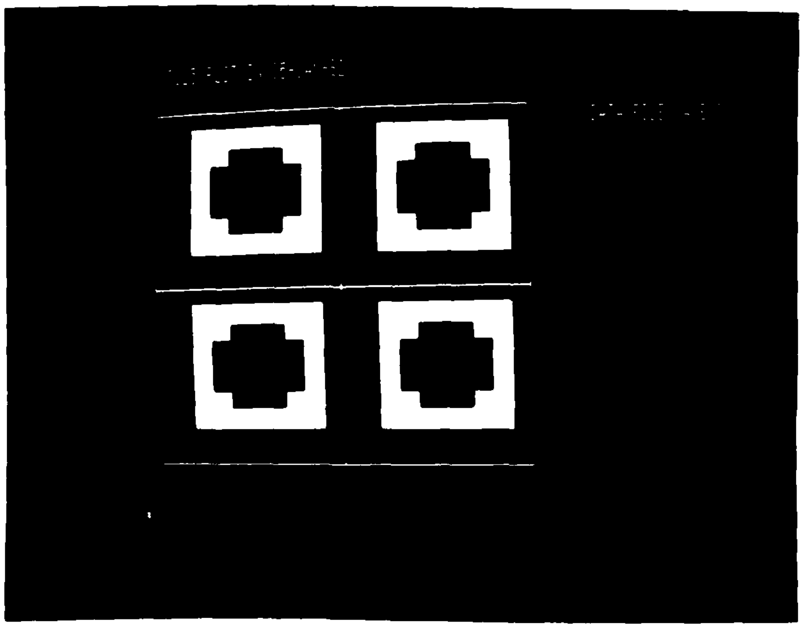


FIG 14. VT125 Chip-level plot of DEMO.DAT wthout individual patterns

7